International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 5, Issue 4, Aug 2015, 55-62
TJPRC Pvt. Ltd.

WEB SERVER PROFILING: APPROACH TO WEBSERVER SECURITY

JYOTI PANDEY & MANOJ JAIN

Department of Computer Science, Haryana, India

ABSTRACT

The Term Web server profiling is an approach to make Websever fully secured from any web attack, whether it can affect any of the web security components like Transmission, System as well as Access Control. This paper presents a program-based intrusion detection system to protect Web server based on the analysis of data source and detection technique using standard Tracing Tool, this paper represents a framework (Methodology, Approaches & Techniques) for monitoring web server activities by using System Tap, and a GUI based Interfacing is done to monitor all three together.

KEYWORDS: Operating System (OS), TCP IP, System Tap, Profiling, Probe, LINUX, Kernel, Tracing, Script Etc

INTRODUCTION

Developing security methods for the Web is a daunting task, in part because security concerns arose after the fact. It offers a survey of Web security issues, focusing on particular areas of concern, such as server security, mobile code, data transfer, and user privacy.

With no insult intended to the early Web designers, security was an afterthought. At the outset, the Web's highest goal was seamless availability. Today, with an internationally connected user network and rapidly expanding Web functionality, reliability and security are critical. Vendors engaged in retrofitting security must contend with the web environment's peculiarities, which include location irrelevance, statelessness, code and user mobility, and stranger-to-stranger communication.

Now a day's Web server security is becoming a very challenging as it includes a wide variety of system including having simple file sharing to the extended critical applications. As nucleus of the web server always must monitor its databases script interpretaters. Securing a web server leads to keep certain factors like remove unnecessary service, remote access, web application content, server side scripting, security patches, auditing of server, removal of un used modules. There are some operating system which releases/support a number of tools to help administrators secure web server installation. Sometimes scanners are also used in a process of securing a web server. Generally to manage World Wide Web server security, on a web server, two levels of security (window security and IIS (Internet Information services)) can be completely integrated, this model allows using authentication based user and grouping.

At OS Level, It can be created various user accounts, configured access permissions for files and directories, and set policies.

THEORY

Web Server basically consists of three essential components System Security (Server security, Host Security), Transmission Security (Data transportation security, Mobile code security) as well as Access control (anonymity and

privacy) [19]120], with all these three components being made robust together, Web server could be treated secure [6]. Typical relations of these components are shown as in next DFdiagrams.

As shown in Figure 1, Security of system is carefully considered under these components. Before launching a specific task experienced attackers often gather information about a target system to exploit a discovered vulnerability.

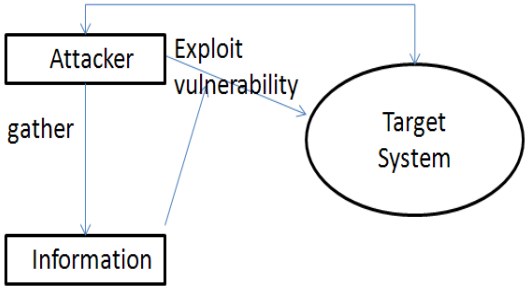

Figure 1

As shown in Figure 2, Transmission security addresses authentication, non-repudiation, integrity and confidentiality of the data communication between transmitter and receiver. Cryptography is an important element of any strategy to cope with those transmission security requirements.

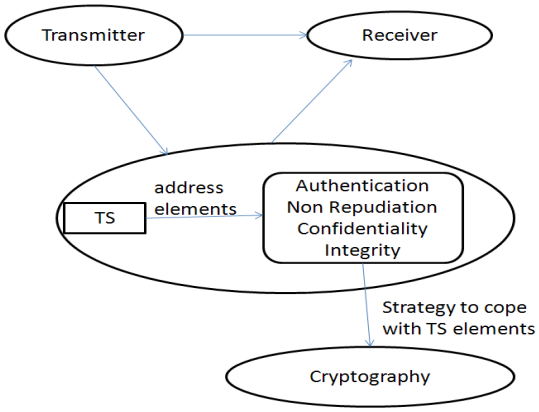

Figure 2

Access control deals with the issue on management of system resources toward system processes and/or users [6][20]. One of the approaches is to using cookies. Cookies were invented to keep continuity and state on the Web. However, sensitive information cannot be securely stored and communicated in typical cookies.

From the Web Server Protection Profile we figured out the minimum-security requirements for a secured Web server used in environments [17] [18]. Then, we have proposed a framework for secured Web server possibly by using System Tap Tool which is helpful for us to develop software for tracing various events of web-server in order to detect any dangerous attack on web server which helps in reducing threat of attacks on web-server in real world.

ANALYSIS [PROFILING]

Web server profiling is being done to protect it from any type of attack threaten with respect to security concern. One of these areas, Network Security, is primarily concerned with keeping unauthorized users from gaining access to the system. The second area, file system security, is concerned with preventing unauthorized access, either by legitimate users

or crackers, to the data stored in the system. The third area, process monitoring deals with process activities and behaviour of process flow in kernel level and looks for effectiveness and the efficiency of processes.

The term Protection is used related to make web server fully secured, just for illustration this concept as if there is some virus present in any system, Antivirus search, detects it and remove it from its all possible existing directories, In same way this software module uses tracing tool System Tap to detect the attack.

Same on web server all data is mostly holded in term of information stored at various kernel level. As any possible attack on web server is possible through either network [TCP IP], through file processing or directory or any input route to network.

SYSTEM TAP [16]

System Tap is a tool that allows developers and system administrators to write and reuse simple scripts to finally examine the activities of a live Linux system. Data can be extracted, filtered, and summarized quickly and safely, to enable diagnoses of complex performance or functional problems.

System Tap scripts are driven by events during execution. There are several kinds of events: entering or exiting a function, a timer expiring, or the entire System Tap session beginning or ending through the use of a kprobe. A kprobe is a Linux kernel tool that is used to interrupt the kernel execution flow even when the system is running. When a probe is reached during execution, a predefined function called a handler can be executed as if it were a quick subroutine then resumes execution to collect debugging information. A probe handler is a series of script language statements that specify the work to be done whenever the event occurs.

This work typically includes extracting data from the event context, storing them into internal variables, or printing well-formatted results. A handler can be made to run before each of these events.

System Tap works by translating its script to a C-language file that is compiled into a Linux kernel module. A Linux kernel module is a binary file containing kernel code that is not loaded into the system during boot but can be dynamically loaded or unloaded from the system. In this case, the Linux kernel module is basically a kprobes module.

When the module is loaded, all probes that are defined in the script are activated and handlers are executed when the instrumented events occur. When the session stops, the probes or hooks are disconnected and the module is unloaded. This behaviour ensures that there is no functional or performance impact when System Tap is not running.

System Tap has an advantage over other debugging techniques due to its powerful scripting language and built-in safety features. Tapsets are libraries of reusable scripts installed with SystemTap that provide helper functions and predefined variables. It can be used in the scripting language and tap sets to create instrumentation code that otherwise require several thousands of lines of kprobes module code when written using C.

System Tap can use Debugging with Attributed Record Formats (DWARF) debugging information that accompanies kernel images and modules, such as how the GDB debugger uses them, to provide fine granular control over debugging. DWARF is a standard for generating information by compilers, assemblers, and linkers that can be used for debugging purposes. Availability of debugging information conforming to the DWARF standard provides visibility into symbols and source code statements.

APPROACH BEHIND SELECTING SYSTEM TAP

There are various tracing tool easily available, but still several remarkable features holded by System Tap, like in terms of safety issues in scripts, as other tracing tool not having much safety concern at a time of infinite loops and recursion, language infrastructure used by system tap is very common to all C Programs, while other tracing tool not .

System Tap Technical Overview

Architecture Overview

System Tap makes kernel debugging in Linux relatively easy, but relies on technology that preceded it to do the work. The most common and well known facility used by System Tap is Kprobes. Another facility, which is growing in use, are kernel markers.

KPROBES

Kprobes is an application programming interface (API) that allows developers to write Linux kernel modules to insert probes into a Linux kernel [12].

Two types of Kprobes are utilized by System Tap- Kprobe and Return Probes.

Kprobe

A Kprobe is a general purpose hook that can be inserted almost anywhere in the kernel code.

Return Probes

A Return probe, also called a kretprobe, attaches to the entry point of a function like a regular Kprobe[12].

Processing Steps

The implementation of System Tap is through the command stap. This command takes as input a file (script) that defines to System Tap where to insert probes and what subroutines to execute when those probes are reached.

Although executing a script with System Tap is as simple as calling the stap command with the script name to run, there are a great number of steps that System Tap go through in order to check the script for errors, convert the script to a form that can be executed in the kernel, run the program, and generate output.

Processing means how system tap gets from a script written by a developer to a kernel module, to producing useful output. Logical flow of system tap processing is shown in DFD, as Figure 3[12][14].

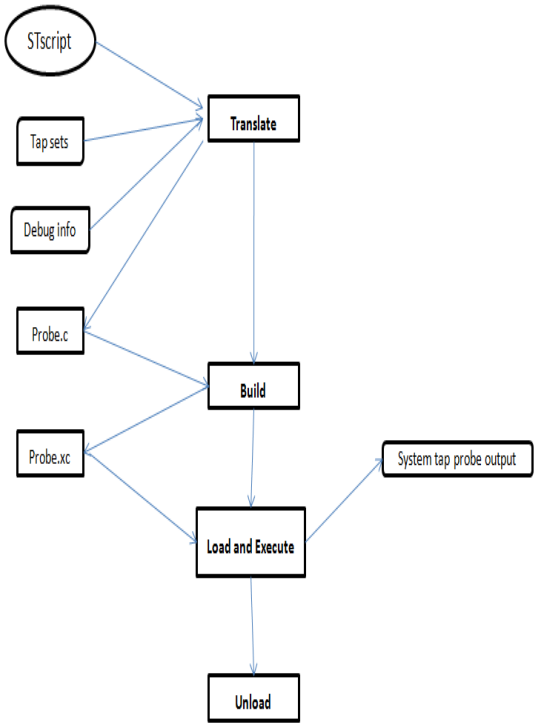

Figure 3

IMPLEMENTATION & METHODOLOGY

Complete Test setup of web server profiling consists three separate part like System Tap [for monitor events], LINUX Architecture [Kernel and hardware] and web server, up to user segments.

This setup has been implemented using System Tap Tool and later on interfaced with java for GUI purpose. Block diagrams shows that System Tap helps to trace events which occur at kernel level of Linux system and web-server hosted at Linux platform and number of users accessing web-server, Test Setup has been shown in Figure 5.

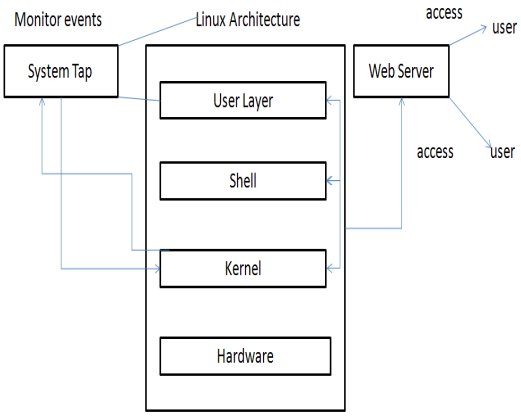

Figure 5

METHODOLOGY SCHEME

Once the test-setup for Web-server Scheme is confirmed with specification implementation of scheme has been

divided into two stages-

- SYSTEMTAP Script
- Interfacing with Java for GUI

SYSTEM TAP SCRIPT

System Tap has various scripts as shown in Figure 4.

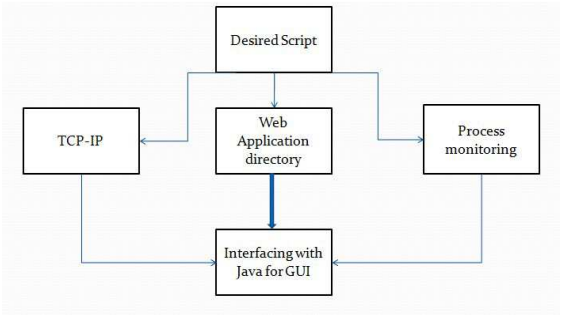

Figure 4

SCRIPT REGARDING TCP-IP CONNECTIONS

The Script monitor (TCP/IP connection) originated at system using kernel function and retrieving result by probing function calls. Main purpose of this script is useful in identifying any unauthorized, suspicious, or otherwise unwanted network access requests in real time.

SCRIPT REGARDING WEB-APPLICATION DIRECTORY

This script describes about I/O activity on specific device where Specific device is Web-Application Directory. This section describes about monitoring files of specific directory if any processes are changing the control access right attributes and processes are changing the attributes of a targeted file in real time.

SCRIPT REGARDING PROCESS MONITORING

This script describes about profiling kernel activity by monitoring function calls. Main purpose is to generate a safe list of processes as a basis for judging the legitimacy of the process.

INTERFACING WITH JAVA FOR GUI

In order to understand the behavior of script output, it has been interfaced with java which will provide GUI for understanding purpose in a complete way. GUI provide options for users to select options regarding displaying number of records, defining script time, defining time for GUI refresh to display latest records on timely basis. Even user is provided with report to view important data for particular day and also having various options for downloading report in .csv format for later use.

PROGRAM MODULES

To make an effective completion of the project, the modules has been divided into various stages like System Tap tool and scripts as shown in Figure 6.

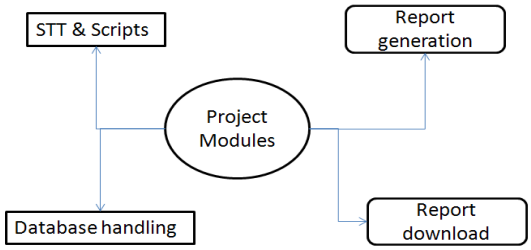

Figure 6

Where scripts are the foundation for web server profiling modules, Report Generation where user allows generating report of various modules on user requests, Report Download where users allow downloading report for later use, and Database Handling where database used for module.

CONCLUSIONS

It presents a framework for monitoring Web Server activities possibly by using System Tap tool. A Program based intrusion and detection system has been successfully implemented, to protect web server from any dangerous attack.

ACKNOWLEDGEMENTS

The authors sincerely thank Dr. Tapas Kumar, HOD CSE Dep.,Lingayas University Faridabad for his encouragement to carry out work. The authors also would like to thank Utkarsh Tyagi , **Abhishek Shukla** for their support, motivation and valuable discussions.

REFERENCES

1. **S.** Jiang, **S.** Smith, and K. Minami, ―Securing Web Servers against Insider Attack," In the 17th Annual Computer Security Applications Conference (ACSAC‘OI j, pp. 265-276.
2. Aviel D. Rubin (AT&T Labs), Daniel E. Geer Jr. Certco,"A Survey of Web Security.
3. A.D. Rubin, D. Geer, and M.J. Ranum, Web Security Sourcebook, John Wiley & Sons, New York, 1997.
4. Jared Karro, Jie Wang," Protecting Web Servers from Security Holes in Server-Side Includes.
5. Sheng-Kang Lin," From Web Server Security to Web Components Security.
6. Christian Gilmore, David Kormann, and Aviel D. Rubin, ―Secure Remote Access to an Internal Web Server," IEEE Network, Vol. 13, Issue 6, pp. 31-37, Nov.lDec., 1999.
7. Jane Curry," Methods of monitoring processes with Zenoss Draft. www.skills-1st.co.uk
8. J.G. Steiner, C. Neuman, and J.I. Schiller, ―Kerberos: "An Authentication Service for Open Network Systems, Proc. Winter 1988 General Conf., Usenix Assoc., Berkeley, Calif., 1988, pp. 191–202.
9. M.K. Reiter and A.D. Rubin, ―Crowds: Anonymous and Web Transactions, ACM Trans. Information Systems Security, Apr. 1998; see alsohttp://www.research.att.com/projects/crowds.
10. Wei Li, Long Chen, Hongjiang Ji, Tong Zhang ,“Improvement of Real-Time Process Monitor Technology on Linux Based on Mandatory Running Control” Vol.4,Issue 4,pp 36-44 , July/August,2000

11. .Bart Jacob,Paul Larsen, Breno Henrique Leitao and Saulo Augusto M Martins da Silva,Instrumenting the Linux Kernel for Analyzing Performance and Functional Problems ibm.com/redbooks.
12. Miles Tracy, Wayne Jansen, Karen Scarfone, and Theodore Winograd," Guidelines on Securing Public Web Servers", September 2007.
13. Don Domingo, Jacquelynn East and William Cohen, "Red Hat Enterprise Linux5 System Tap Beginners Guide".2011.
14. Installing and Using System Tap, IBM Corporation 2008, 2009.
15. Josh Stone, "Dynamic Tracing and Performance Analysis Using System Tap" Software Engineer Software and Solutions Group August 4, 2008.
16. "Web Server Protection Profile (Draft Version: .6j," July 31, 2001, http:llniap.nist.govIcc-scheme/PP~WSPP~VO.6.Html.
17. Joon **S.** Park and Ravi Sandhu, "Secure Cookies on the Web," in the IEEE Internet Computing, Vol. **4,** lssue **4,** pp. 36-44, lul./Aug., 2000.
18. Joon **S.** Park, Ravi **Sandhu,** and Gail-Joon Ahn,"Role-Based Access Control on the Web," in the ACM Transaction on Information and System Security, Vol. 4, No. 1, pp. 37-71, Feb.,200 1.